

AN AUTOMATIC FRAMEWORK FOR EMBRYONIC LOCALIZATION USING
EDGES IN A SCALE SPACE

A Thesis
Presented to
The Faculty of the Department of Computer Science
Western Kentucky University
Bowling Green, Kentucky


In Partial Fulfillment
Of the Requirements for the Degree
Master of Science

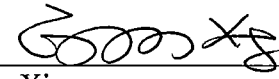
By
Zachary Bessinger

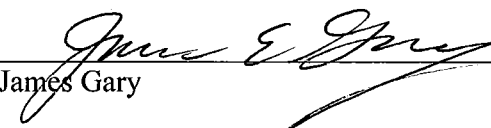
May 2013

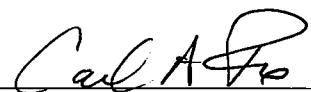
AN AUTOMATIC FRAMEWORK FOR EMBRYONIC LOCALIZATION USING
EDGES IN A SCALE SPACE

Date Recommended 04/03/13


Qi Li, Director of Thesis


Guangming Xing


James Gary

 5-6-13
Dean, Graduate Studies and Research Date

ACKNOWLEDGMENTS

I would like to thank my thesis adviser, Dr. Qi Li, for giving me the opportunity to work on research in this field while in my baccalaureate. The continued support and knowledge in my graduate studies were integral to piquing my interest in this subject and motivating me to develop this work. He has given me opportunities that I don't think I would have had otherwise, and for that I am grateful.

I would also like to thank my committee members, Dr. Guangming Xing and Dr. James Gary, for reviewing my thesis document and listening to my defense. I'm glad to have their support for creating and defending my thesis.

I would like to acknowledge others who have helped me along the way in developing my thesis, when writing and in down time. Specifically, I'd like to thank Mitchell Michael for offering ideas to improve my techniques. I'd like to thank those who have had confidence in me over the years, helping me relax when I was in deep need of it, and thus preparing me for more thesis writing and development. Though there are many unnamed thanks, those who read this will know who they are. Finally, I'd like to thank my family that have given me nothing but the best throughout my whole life, which made this whole experience possible.

TABLE OF CONTENTS

| | Page |
|--|-----------|
| 1 INTRODUCTION | 1 |
| 1.1 Background | 1 |
| 1.2 Motivation | 3 |
| 1.3 Contribution of the Thesis | 4 |
| 1.4 Structure | 4 |
| 2 BACKGROUND AND RELATED WORK | 5 |
| 2.1 Basic Techniques | 5 |
| 2.1.1 Gaussian Scale Space | 5 |
| 2.1.2 Edge Detection | 6 |
| 2.1.3 Connected Components | 7 |
| 2.2 Existing Work on Embryo Localization | 8 |
| 2.2.1 Segmentation | 8 |
| 2.2.2 Shape Modeling | 9 |
| 2.2.3 Active Contour | 10 |
| 2.3 Advanced Versus Low-Level Techniques | 10 |
| 3 PROPOSED FRAMEWORK | 12 |
| 3.1 Scale Space Framework | 12 |
| 3.1.1 Gaussian Smoothing | 12 |
| 3.1.2 Scale Space | 13 |
| 3.1.3 Edge Detection | 14 |
| 3.1.4 Connected Components | 16 |
| 3.1.5 Convex Hull | 18 |
| 3.2 Scale Selection Criteria and Constraints | 19 |
| 3.2.1 Criteria on Optimal Scale Selection | 19 |
| 3.2.2 Constraints | 21 |

| | | |
|----------|--|-----------|
| 3.3 | Results | 26 |
| 3.4 | Conclusions | 28 |
| 4 | IMPROVEMENT OF LOCALIZATION EFFICIENCY | 30 |
| 4.1 | Scale Range Reduction | 30 |
| 4.1.1 | Motivation | 30 |
| 4.2 | Construction of the σ_L – σ_H Table | 36 |
| 4.3 | σ_L to $[\sigma_{H,min}\sigma_{H,max}]$ Mapping | 37 |
| 4.4 | Up-scaled Low-scale Contours Versus Original High-scale Contours | 40 |
| 5 | CONCLUSION | 43 |
| 6 | BIBLIOGRAPHY | 45 |

LIST OF FIGURES

| | | |
|-----|--|----|
| 1.1 | <i>Drosophila</i> embryos with various gene expression regions (blue/indigo coloring) and conditions. Left to right, top to bottom, (A), (B), (C), and (D) . . . | 2 |
| 2.1 | 4-connected and 8-connected pixel windows | 8 |
| 3.1 | Scale space representation of embryonic images over the family $2^i, i \in [0, 8]$ | 14 |
| 3.2 | Canny edge detector with heuristic thresholding using scale parameter $\sigma = 16$ | 16 |
| 3.3 | Connected components located in an embryonic image | 17 |
| 3.4 | Convex hull (in red) of a binary image (courtesy of Mathworks) | 18 |
| 3.5 | Effect of applying the angular checking constraint with $a_i \geq 135^\circ$ | 22 |
| 3.6 | Sample images selected without border checking constraint (on left) with corrected image using the border checking constraint (on right). | 24 |
| 3.7 | Effects of applying each criterion on embryonic images with varying conditions. | 25 |
| 3.8 | Successful localizations output from the framework. | 27 |
| 3.9 | Failed localizations from the framework [4] | 28 |
| 4.1 | Framework results on a set of ISH images at original resolution with $\sigma \in [1, 40]$ | 31 |
| 4.2 | Computation time of a (raw) embryonic image exponentially decreasing with respect to scale values | 33 |
| 4.3 | Computation time with respect to changes in resolution | 34 |
| 4.4 | Table range of values for $\sigma_L \rightarrow [\sigma_{H,min}, \sigma_{H,max}]$ | 37 |
| 4.5 | Table mapping distribution for $\sigma_L = 1 \rightarrow \sigma_H \in [1, 40]$ | 38 |
| 4.6 | Table mapping distribution for $\sigma_L = 1 \rightarrow \sigma_H \in [2, 40]$ | 39 |
| 4.7 | Embryonic image overlayed with low and high resolution contours. | 41 |

AN AUTOMATIC FRAMEWORK FOR EMBRYONIC LOCALIZATION USING EDGES IN A SCALE SPACE

Zachary Bessinger

May 2013

46 Pages

Directed by: Qi Li, Guangming Xing, and James Gary

Department of Computer Science

Western Kentucky University

Localization of *Drosophila* embryos in images is a fundamental step in an automatic computational system for the exploration of gene-gene interaction on *Drosophila*. Contour extraction of embryonic images is challenging due to many variations in embryonic images. In the thesis work, we develop a localization framework based on the analysis of connected components of edge pixels in a scale space. We propose criteria to select optimal scales for embryonic localization. Furthermore, we propose a scale mapping strategy to compress the range of a scale space in order to improve the efficiency of the localization framework. The effectiveness of the proposed framework and the scale mapping strategy are validated in our experiments.

INTRODUCTION

1.1 Background

Over the course of the last century, with the advent of many scientific and technological advances, biologists have made many advances in studying the effects of gene interactions and their relationships. The impact of this is visible when referring to such projects as the Human Genome Project [7], which is actively attempting to sequence the entire human genome. To understand our own genome, we build models based on creatures whose genetics are not as intricate, such as insects. Biologists have chosen to study *Drosophila* (fruit fly) embryos because of their rapid life cycle and limited genetic sequence. The Berkeley *Drosophila* Genome Project (BDGP) was started at the University of California, Berkeley as a consortium between genome and cancer research organizations to study and document the many genetic variations of *Drosophila* embryos given their stage of development, potential mutations, and gene-gene interactions [1].

One component of the BDGP involves finding spatial patterns of gene expression in *Drosophila* embryogenesis through the use of *in situ* hybridization (ISH) [19]. The *in situ* hybridization technique involves localizing a specific DNA or RNA sequence in tissues using the labeled strand complementary to the target pattern (probe). After undergoing this technique, the embryos are left with certain areas that may or may not be stained depending upon the probe's reaction. The resulting stained patterns found in the embryos show when (the stage of development) and where the target gene is expressed. Scientists believe that understanding the pattern formations and the roles of certain genes in an organism will lead to advances in medicinal treatment, such as genetic therapy [22, 24].

Figure 1.1 demonstrates the majority of the variations that can occur in the *Drosophila* ISH images. The blue or purple regions on the embryos are the gene expression regions. These images can come in a variety of forms. Some of the variations

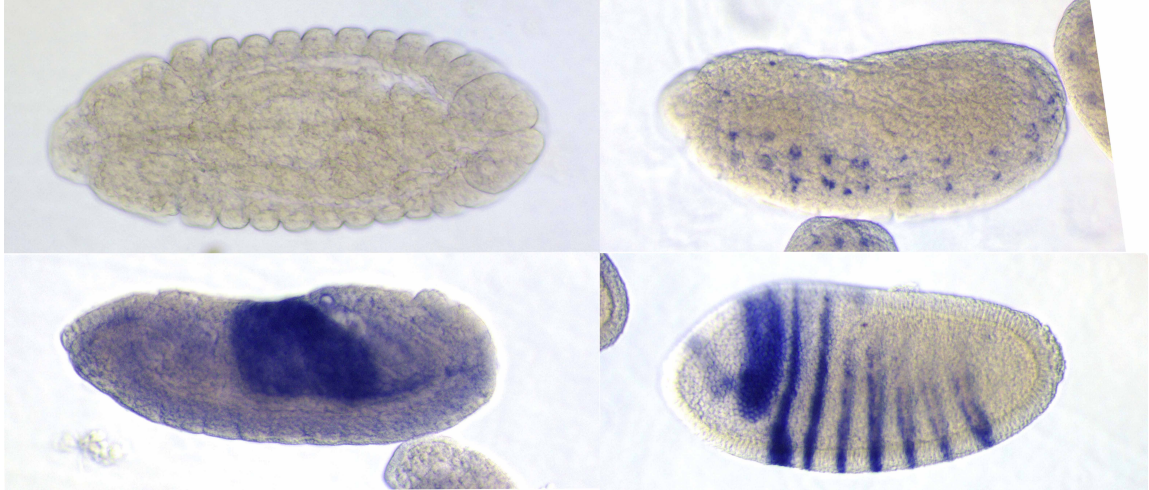


Figure 1.1: *Drosophila* embryos with various gene expression regions (blue/indigo coloring) and conditions. Left to right, top to bottom, (A), (B), (C), and (D)

that can occur in these images are due to factors such as gradient, blur, and color. We consider these features that relate to the image itself as internal factors.

There are also external factors of the image that contribute to an ISH image. We use two phrases to define the external factors of embryonic images. In the case where there are multiple embryos in an image, we consider this as a partial embryo because there are other embryos in the image that exist aside from the target embryo. The partial embryos can also be called touching if they are located on the contour of the targeted embryo. A touching embryonic image contains a partial embryo, but not necessarily vice-versa. The first image (A) in Figure 1.1 is an embryo that is later in stage of development due to the muscle development along the outermost part of its body. We consider it to be an optimal image due to its sharpness, low color variation, and lack of partial/touching embryos. The (B) and (C) images in Figure 1.1 show occurrences of touching embryos. Image (D) in Figure 1.1 shows a partial embryo that is not considered to be touching. Some further examples of external factors include strong expression region striations and developmental stage shape.

1.2 Motivation

As of this writing, the BDGP has documented more than 111,000 *Drosophila* images along with more than 7500 gene expressions with very limited computer aid. Due to the sheer quantity of embryonic images, it is necessary to design a system to aid in the classification of the numerous variants of these embryonic images and process them as precise as possible. Creating an embryonic ISH image processing framework is a non-trivial task due to the many internal and external factors.

Before we can begin to analyze an embryonic image, we need some way to perform the localization. Localization is one of the first steps that must be taken before performing analysis in order to properly pinpoint the targeted embryo. Manual techniques currently exist for localizing a specific embryo in the image, however they involve human interaction to manually click and generate a set of points along the contour of the targeted embryo to perform the localization.

Though the BDGP has been in development for some time, the project faces difficulties when it comes to analyzing the embryos. In order to assure the classification validity, experts on the *Drosophila* are needed to analyze expressive regions of the embryonic image after reacting to some gene. It is a time consuming process, which requires experts to have to perform this annotation themselves. The time needed to locate a particular region and assess the embryo could be better spent if there were a system that can automatically and accurately identify the contour information of the embryonic images. This creates a strong need for a framework which can automate the process of localization and recognition of the *Drosophila* images to alleviate the burden on the scientists. Despite the immediate need for automation, little research has been performed to design such a system.

1.3 Contribution of the Thesis

In this thesis, we propose a framework to determine the optimal localization by analyzing the connected components in Lindeberg’s scale space [16] using low-level image processing features such as filtering and edge detection. Our framework uses a sequence of low and high-pass filtering combined with connected component analysis to perform the localization. We propose a criteria-based approach to systematically determine an optimal criterion for selecting the most precise localized contour. We also improve the localization accuracy of the framework by adding constraints and proposing an inter-resolution mapping scheme as a means to improve efficiency and reduce the overall computation time. This thesis focuses on the development of the proposed framework from its theory and inception to the extensions and constraint modifications to improve accuracy and efficiency.

1.4 Structure

The organization of the thesis is as follows: Chapter 2 provides a literature review of the methods and background of the techniques applied in our framework. Chapter 3 provides detail of the criteria-based selection methods utilized, including additional constraints to improve the accuracy and the optimal criterion. Chapter 4 presents the inter-resolution mapping scheme to reduce the scale space parameter search and the results of the mapping. Chapter 5 gives our conclusions based on the results and future experiments based on this research.

BACKGROUND AND RELATED WORK

In this chapter, we will first present basic techniques (Gaussian scale space, edge detection, and connected components) that will be used to develop a localization framework. Then, we review existing works on the localization of embryos that can be categorized into three schemes: i) segmentation, ii) shape modeling, and iii) active contour.

2.1 Basic Techniques

2.1.1 Gaussian Scale Space

The Gaussian scale space is defined through the convolution of an image with a Gaussian kernel over a family of convolutions with varying scale parameters. For use in images, we define the Gaussian kernel, G , in terms of two dimensions, defined by the equation:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (2.1)$$

where σ refers to the width (size) of the Gaussian kernel. It's important to note that σ should be odd; so that when the convolution is performed each pixel is multiplied with the Gaussian kernel. In terms of image processing, we can define the convolution of an image and a kernel as the dot product of a kernel rotated 180 degrees centered on each pixel in I . Once the kernel is centered we can simply use the equation:

$$I_\sigma = I \cdot G(x, y) \quad (2.2)$$

In the case where σ is even, we simply add one to ensure the kernel is odd.

Since the Gaussian scale space is the set of convolutions over a given parameter range, we can redefine the equation for generating a Gaussian kernel in terms of t , where

$t = \sigma^2$ yielding:

$$G(x, y; t) = \frac{1}{2\pi t} e^{-\frac{x^2+y^2}{2t}} \quad (2.3)$$

When we convolve images based on a range of values for t , we generate a set of images that is called the scale space representation of an image. As t increases, the image becomes successively smoother after each convolution.

Once we have smoothed images, we can begin to find the edges. Smoothing is a critical step to reduce the amount of noise in an image (false edges) and achieve some optimality in terms of correct edges. To find the edges within an image we can use one of the many variants of edge detectors.

2.1.2 Edge Detection

There are many different kinds of edge detectors such as Roberts, Prewitt, and Sobel [20]. Though there's an array of edge detectors to choose from, the most optimal edge detector in terms of preserving good edges and removing bad edges is the Canny edge detector, which is the most popular variant of edge detection schemes. Researchers use the Canny edge detector because of the three characteristics that motivated Canny's detector [6].

1. Good detection – Low probability of spurious edges, which involves maximization of the signal-to-noise ratio.
2. Good localization – Edge points should be accurate to the true edge.
3. Single edge response – Given two responses on a single edge, one must be false.

One of the initial steps in the Canny edge detector is convolving the gray-scale representation of the image with a Gaussian filter. The σ value in the Gaussian filter determines the amount of noise reduced from the image, or variation of color or brightness

in a given image. We can use the gradient of an image, denoted ∇I , to compute the maximal change in either direction. Once the image has been smoothed with the Gaussian filter, we use the gradient images to examine the partial derivatives for maximal change in both the x and y direction, denoted by ∇I_x and ∇I_y respectively. Once we have the gradient images, we use non-maximum suppression to reduce the number of wide edges found in an image. Non-maximum suppression first considers an angle between a particular band of edge pixels. Once the angle has been located, we select the edge to be the closest to the middle of this region. We reduce the width of the band to one pixel based on the angle and its closeness to other pixels. By reducing the edge to one pixel, we reduce the size of the edge itself to a single, accurate pixel. This in turn reduces the size of the edge set allowing for computations to be performed on a smaller edge pixel set. Since the edge pixel set is most likely non-continuous we perform extra analysis or morphological operations on the image.

2.1.3 Connected Components

The idea of connectivity and connected components originate from basic graph theory, and refers to the minimum number of vertices to be removed such that all the others remain connected. In images, connected components are blobs of pixels defined by a pixel neighborhood. Images can have connected components in two or three dimensions, however in most cases connected component analysis takes place in two dimensions. When locating connected components in two-dimensional images we either use a 4-pixel or 8-pixel neighborhood with respect to the center pixel to determine if the pixel belongs as a member of the component.

Connected components are utilized for various purposes in computer vision and image processing research for image segmentation and labeling. For the purposes of this research, we analyze the connected components to label them and perform region of

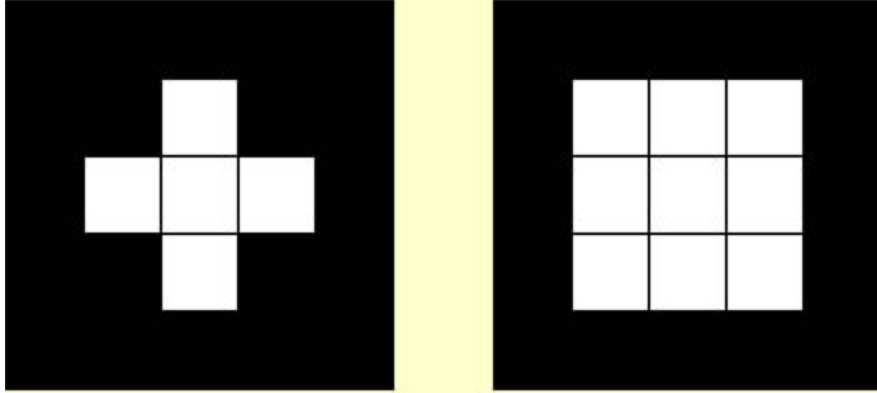


Figure 2.1: 4-connected and 8-connected pixel windows

interest (ROI) extraction.

2.2 Existing Work on Embryo Localization

In the research that has been attempting to localize an embryo, most of the techniques they use involve exploitation of the embryonic conditions. In previous research, there are three popular schemes:

1. Segmentation
2. Shape modeling
3. Active contour

2.2.1 Segmentation

Segmentation refers to dividing an image into multiple sets of pixels. In medical and biological imaging, image segmentation is one of the most commonly used techniques [23]. We can categorize the segmentation into two types:

1. Region-based

2. Edge-based

Region-based techniques take into account the internal qualities of an image, such as color, luminosity, etc. Region-based techniques often use connectivity to perform the segmentation, and have been used in previous research to segment the background from the foreground of an embryonic image. These techniques have potential to work in the case of *Drosophila* images due to the nature of the ISH images typically having a significantly different background from foreground and have been implemented in much of the earlier research [9, 22]. Naturally, these region methods also lead into using edge-based methods.

Recall that edge based techniques observe high gradient changes along a direction. Since edge detection involves high pass filtering and outputs an unorganized edge pixel set, there is extra preprocessing (smoothing) and postprocessing (in Canny edge detector, hysteresis) that must occur. The region and edge-based techniques are relatively low level, however more advanced techniques have been used to achieve segmentation. One frequently used advanced method that has been utilized in contour extraction is the watershed transform [5], which can be performed in various ways, but essentially works through flooding the gradient image. The watershed transform has been widely used as the primary segmentation method in localizing the *Drosophila* embryos and other forms of bioinformatics research [10, 18, 21].

2.2.2 Shape Modeling

When dealing with the embryonic images, research has been extended to use either an elliptical model [21] (since the embryo is somewhat of an ellipsoid) or Eigen-shape model to fit the contour of the ellipse [14, 17]. Elliptical modeling may be beneficial when dealing with images known *a priori* to contain elliptical targets, however the detection time for ellipses can be time consuming caused by the need for many

parameters [8]. Eigen-shaping may be more effective than elliptical modeling, however it also depends on the context of the target. To properly use Eigen-shaping we should know the targeted image, which takes away from the generalization. However, if we do not know the proper shape of the target, we can use other techniques such as active contour.

2.2.3 Active Contour

Active contour is a technique introduced in Kass et al [11] that, over a sequence of iterations, attempts to fit the contour of the target object [26]. Active contour is widely used in medical imaging and even in the particular case of contour extraction of *Drosophila* embryos [2, 3, 18, 25]. Active contour is highly dependent upon the parameters used to accurately localize the contour of a target. Unfortunately, active contour requires so many parameters that finding an optimal set of parameters for an image is non-trivial. If a parameter set is found and tuned properly, the technique is highly accurate but still suffers from high computation time. Typically, these parameters can be adjusted using more advanced statistical techniques.

2.3 Advanced Versus Low-Level Techniques

In order to achieve the accuracy levels of the previous research, they used deep statistical analysis through use of clustering or building statistical models, such as principle component analysis (PCA), linear discriminant analysis (LDA), or Gaussian Mixture Models (GMM). Even by using more complex statistical techniques, the experiments in the previous research yielded imprecise results. We believe that with our framework we can use low-level techniques combined with minimal heuristics to achieve better results than those performed using the advanced statistical analysis and modeling. Since the contour of an embryo aids in determining the stage, localization precision is a key aspect of any viable framework. We believe the experiments involved in other

research lack accuracy simply due to the localization problem. In the next chapter we describe the framework and provide an explanation of the methods we used to construct our framework.

PROPOSED FRAMEWORK

In this chapter, we propose an automated framework for localizing regions of interest (ROI) of embryonic images using low-level techniques through analysis of the connected components within the Gaussian scale space¹. The proposed framework is based on low-level image processing techniques, such as filtering and edge detection. One of the key issues in the proposal framework is on how to select the optimal scale. Specifically, this chapter consists of two sections: Section 1 presents the main structure of the proposed framework, and Section 2 presents three criteria used to select optimal scale in order to achieve optimal localization of the contour of the targeting embryo in an image.

3.1 Scale Space Framework

3.1.1 Gaussian Smoothing

The Gaussian filter plays an integral role in the establishment of this framework. We use the Gaussian filter for many different purposes, and its output is input to nearly every other technique we use. The Gaussian filter is used to generate the scale space representation of the embryonic image. We use the Gaussian filter because of its many properties, those of which define the scale space under the axioms of linearity, scale invariance, shift invariance, etc. Since these are well-established conditions, it is relatively standard to use only the Gaussian filter as of this time. The convolutions between an image and a Gaussian filter uses zero-padding on the border such that when a kernel is near the border and there is overlapping space that runs outside of the boundary, we assume the out of bounds pixel region values are zero. The Gaussian kernel is a prerequisite as input to the Canny edge detector.

¹Preliminary result has been published in ICIP '12 [4]

3.1.2 Scale Space

Recall that the Gaussian scale space representation of an image is the convolution of a sequence of Gaussian filters of varying σ to create a multi-scale representation of the same image. Lindeberg popularized the development of scale space techniques and scale selection [15]. To perform the scale selection, Lindeberg proposed utilization of the γ -normalized derivatives to achieve a one-to-one correspondence between the responses of a sequence of Gaussian derivative kernels with an image. The next step is to maximize the response, which results in preservation of feature points across various scales.

In our framework, we utilize a variant Lindeberg’s scale space theory. If we were looking for a sequence feature points, finding them using correspondence between the scale space images would be appropriate and has been utilized in such techniques as image pyramids. Rather than search for feature points as the idea behind our scale selection, we are interested in the optimal response that can generate a “good” contour. This means we seek not only the feature points, but also the underlying topology that the target object possesses. The scale space representation of the embryos is then passed as input into the edge detector. Using the connected component analysis as described in the previous section, we propose a criteria-based approach to select an optimal contour.

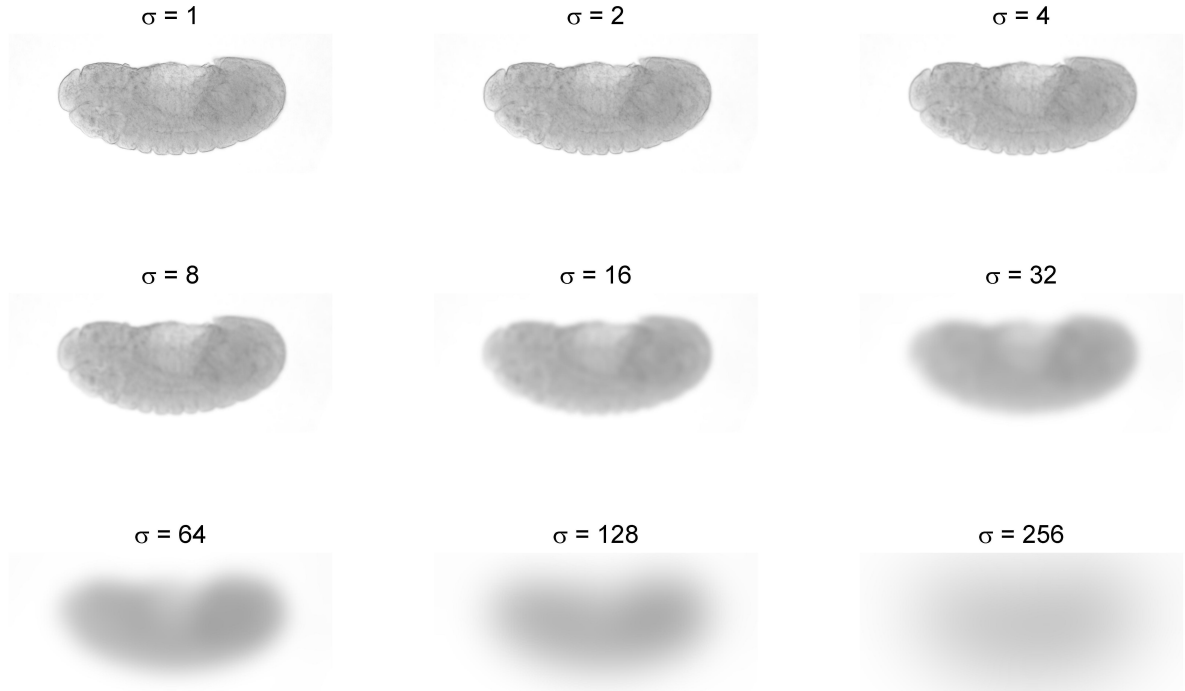


Figure 3.1: Scale space representation of embryonic images over the family $2^i, i \in [0, 8]$

3.1.3 Edge Detection

As previously mentioned, an essential step before applying a high-pass filter is to use a low-pass filter to smooth an image. We choose to use the Canny edge detector rather than other edge detectors because of its wide use and acceptability as a “good” edge detector. Recall that the Canny edge detector requires a minimum and maximum threshold for use in the hysteresis step to determine the edges to keep or discard. In our experiments, the threshold parameters are done heuristically based on the image itself. To select the threshold values, we follow the Canny detection process that involves computing the magnitude of the gradient image, ∇I , by addition of the gradient in the x and y -directions, denoted ∇I_x and ∇I_y .

$$\nabla I_x = \frac{\partial I}{\partial x} \hat{x} \quad (3.1)$$

$$\nabla I_y = \frac{\partial I}{\partial y} \hat{y} \quad (3.2)$$

$$||\nabla I|| = \sqrt{\nabla I_x^2 + \nabla I_y^2} \quad (3.3)$$

Before we convolve the gradient image with the Gaussian filter, we should normalize the magnitude of the gradient. Note that the gradient image is the same size as the input image, $m \times n$. We also define a value, p , relating to the percent of pixels in the image that shouldn't be selected as edges. This value sets an upper bound for the allowable number of edge pixels and ensures reduction in the edge pixel set. For our experiments, we assign $p = 0.7$.

Denote the threshold value, t , and the range of low and high threshold values as $[t_{low}, t_{high}]$. We normalize the image to generate the gradient magnitude image and ensure the threshold values in the range are $t \leq 1$. Once we have the gradient magnitude image, we categorize the pixels into 64 buckets. Given a bucket b in the set of buckets B , we calculate the high threshold by $\forall b \in B, \sum b > (m \times n \times p)$, then we take the cumulative sum of b and divide it by the number of buckets, which in our case is :

$$\frac{\sum b}{64} \quad (3.4)$$

and we set this value as our high threshold. The low threshold is computed based on the value returned from calculating the high threshold. In our experiments, we assign the low threshold to 40% of the high threshold.

After running the detector using this heuristic threshold, it tends to yield optimal results in terms of what defines a “good” edge detector as described in Chapter 2. Now

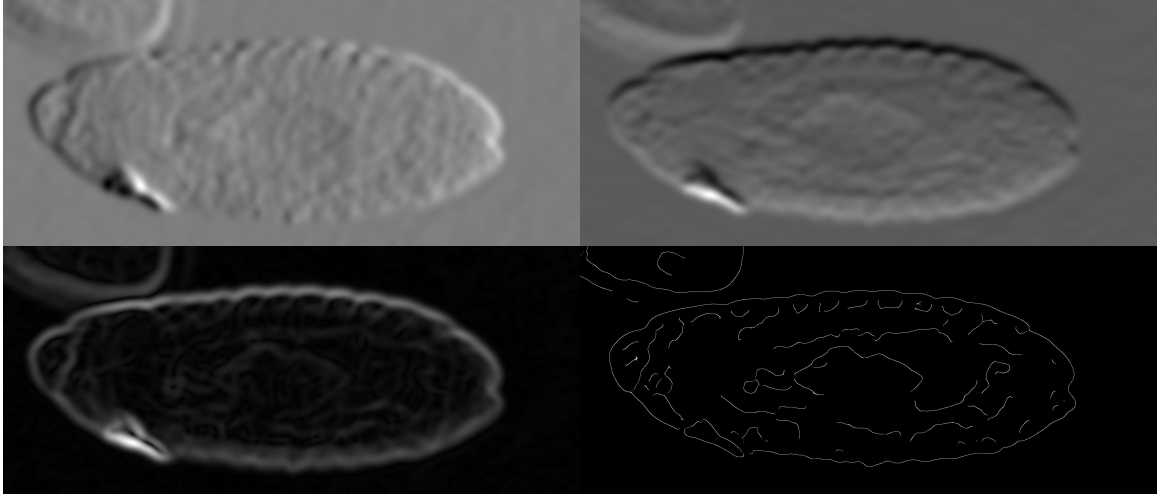


Figure 3.2: Canny edge detector with heuristic thresholding using scale parameter $\sigma = 16$

that we have the edge information of the image, we must perform addition operations since the edge pixels are essentially unordered and unlabeled. The next step we take is to partition the edge pixels into a series of components based on their relativity to other edge pixels within a neighboring region. We do this using a technique called connected component labeling.

3.1.4 Connected Components

Connected components can be located within both two and three dimensions, but in this research we operate on binary (two-dimensional) images. The input and output of the connected component analysis both use binary images. In particular, our binary image is an edge image whose edge pixel set has been transformed into component regions by an 8-connected pixel neighborhood.

Let $C_{i,\sigma}$ represent a connected component at a scale value σ . We also denote E_{I_σ} as the binary image that contains a set of connected components, mathematically defined as :

$$E_{I_\sigma} = \bigcup_i C_{i,\sigma}, \quad (3.5)$$

$$C_{i,\sigma} \cap C_{j,\sigma} = \emptyset, i \neq j, \quad (3.6)$$

such that $C_{i,\sigma}$ refers to the set of connected components at a particular scale, σ . Each connected component in E_{I_σ} is unique, however given two scales σ_1 and σ_2 , we cannot guarantee the relationship $C_{i,\sigma_1} \subseteq C_{i,\sigma_2}$ or $C_{i,\sigma_1} \supseteq C_{i,\sigma_2}$.

We are also interested in the largest connected component in an image. Two of our three criteria involve analysis of the largest connected component within a connected component image, E_{I_σ} . We denote the largest connected component at scale σ as :

$$i_\sigma = \operatorname{argmax}_i |C_{i,\sigma}| \quad (3.7)$$

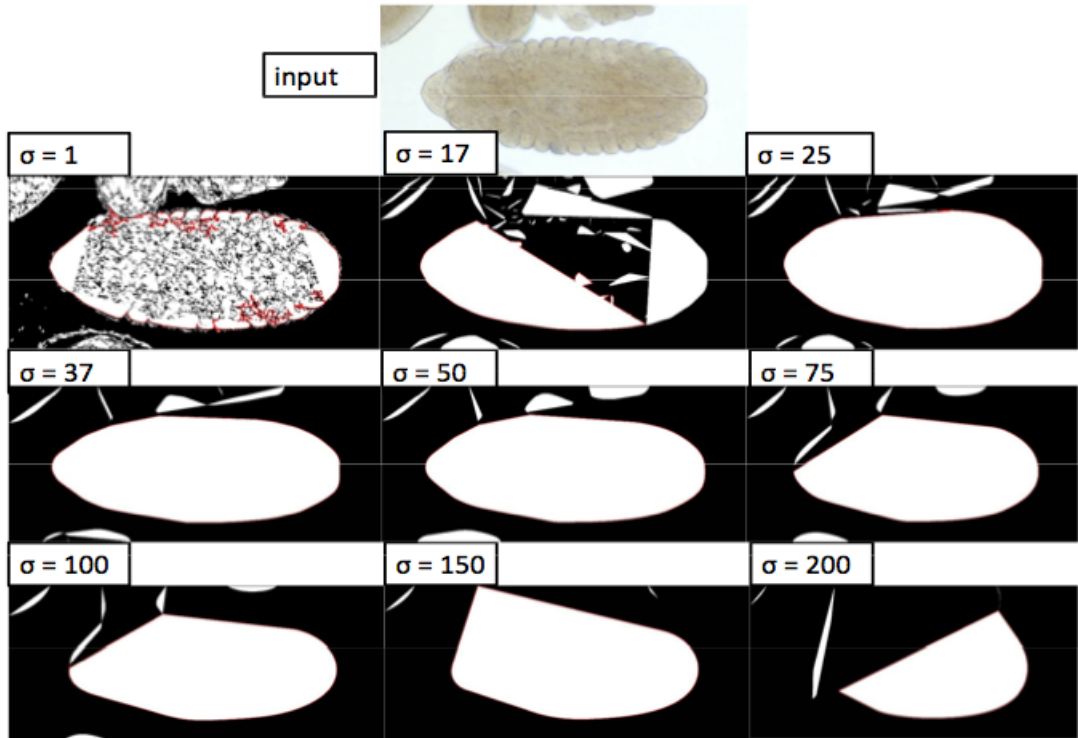


Figure 3.3: Connected components located in an embryonic image

While locating the connected components will produce a series of connected component image, extra analysis is needed to ensure an accurate shaping of the contour.

3.1.5 Convex Hull

Our framework also utilizes the convex hull for localization precision and smoothing. Given an edge image, E , the convex hull of the image is defined as the unique set of pixels in E that is the smallest convex set containing E . Informally, one may envision the convex hull as stretching a rubber band around a set of points. The rubber band then is representative of the convex hull of that set of points.

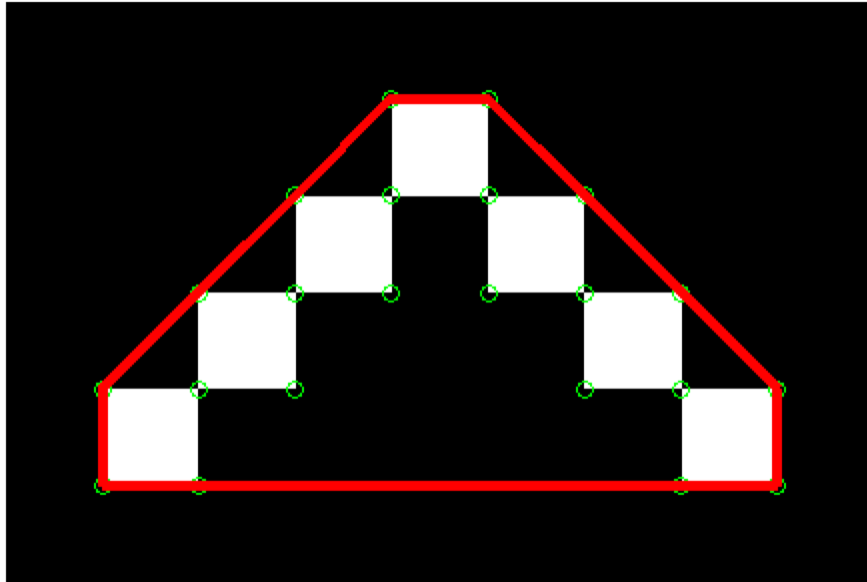


Figure 3.4: Convex hull (in red) of a binary image (courtesy of Mathworks)

In our framework, we use the convex hull in the generation of connected components. Each of the edge images in Figure 3.4 has the convex hull taken of each connected component within an 8-connected pixel region.

3.2 Scale Selection Criteria and Constraints

The primary factor for determining the criteria to experiment originates from the reaction of the connected components over the scale parameter space. Upon inspection of Figure 3.3, we can notice how the connected components have a tendency to merge and diffuse. The merging process reduces the quantity of connected components and increases the size of the central, larger component. The diffusion process occurs when further merging cannot occur, so the largest component begins to shrink. These trends shape the three criteria we propose, which are:

- (a) Minimization of the number of connected components
- (b) Maximization of the largest connected components
- (c) Minimization of shape inconsistency

3.2.1 Criteria on Optimal Scale Selection

Minimization of the Number of Connected Components

At low value scales, such as $\sigma = 1$, the embryo tends to contain too much detail and as a result becomes over-segmented. As the scale increases, more and more connected components gain area and tend towards each other. Over time this process continues until the number of connected components becomes relatively low and stable. The criterion to minimize the number of connected components is defined as:

$$\sigma^* = \operatorname{argmin}_{\sigma} |\{C_{i,\sigma}\}|, \quad (3.8)$$

where $|\{C_{i,\sigma}\}|$ indicates the number of connected components. Note that this minimization step typically occurs in images that contain partial or touching embryos.

Embryonic images that have only a single embryo are not subject to outlying connected components for this reason. In the case where $|\{C_{i,\sigma}\}|$ remains constant over the entire scale range, we select the minimal scale value. This will ensure that the component is large and that over-segmentation is less likely to occur.

Maximization of the Largest Connected Components

Rather than minimize the number of connected components, we want to analyze the largest components across the entire scale range. Noticing the merging/diffusion trend, we believe that the optimal contour can be achieved at the point where after merging the central component reaches a peak area before diffusion. We define the maximal area over all the connected component images as :

$$\sigma^* = \operatorname{argmax}_{\sigma} |\{C_{i,\sigma}\}|, \quad (3.9)$$

The risk of partial embryos to creating discrepancies between itself and the target embryo necessitates us to add constraints to ensure that we only select a contour that belongs to the target embryo. Partial embryos may be close enough that they are not touching, however have an effect on the contour generated by the edge detector. For this criterion, we apply a constraint on the cross angle between a set of subsampled pixels. The angular checking constraint is described in more detail in Section 3.2.2.

Minimization of Shape Inconsistency

This criterion aims to minimize the inconsistency between a standard ellipse and the embryo. The motivation of this criterion originates from the fact that the embryo itself is somewhat elliptical shaped. To perform the fitting, we use the direct least-squares fitting method described by Fitzgibbon et al [8]. This method involves modeling the standard equation of an ellipse

$$S(p) = ax^2 + by^2 + cxy + dx + ey + f = 0, \quad (3.10)$$

to a point set, denoted P . Given a point $p(x, y)$, where $p_i \in P, \forall p_i$ we estimate the parameter values for a, b, c, d, e , and f by linear fitting using Eigen-decomposition, as described in [8]. The ellipse we're fitting is fitted relative to the largest connected component, $C_{i_\sigma, \sigma}$.

To validate the fitness of a standard ellipse to an embryo, we use the point to set difference, that is $d(C_{i_\sigma, \sigma}, S)$, calculating the distance from each point in the largest connected component and the ellipse. The goal of this criterion is to minimize the shape inconsistency, which we express as:

$$\sigma^* = \operatorname{argmin}_\sigma d(C_{i_\sigma, \sigma}, S), \quad (3.11)$$

Since we are trying to minimize shape inconsistency, the inconsistency is found through maximizing the distance from the estimated shape using the worst-case scenario.

3.2.2 Constraints

Angular Checking

Observing Figure 3.2 and examining the largest component in each of the images, we can see the merging and diffusion of the smaller connected components towards and away from the largest connected component. If we take the scale space parameter range from the left and analyze the progression, we can see that over a period of time the sharp edges from the largest connected component are strongest for the lowest and highest scale range values. If the largest component has sharp edges, it should be rejected due to the natural shape of the embryos. There are no *Drosophila* embryos that have sharp corners, so introducing an angular checking constraint is an intuitive approach

to eliminating invalid candidate contours. The angular checking constraint subsamples pixels and examines the angle between two sequential pixels. Given the largest connected component, $C_{i\sigma,\sigma}$, denote $c = \{p_0, p_1, \dots, p_n, p_0\}$ as the boundary of the connected component $C_{i\sigma,\sigma}$. Also denote $a_i = \cos^{-1} \left\langle \frac{p_{i+1}-p_i}{\|p_{i+1}-p_i\|} \middle| \frac{p_{i-1}-p_i}{\|p_{i-1}-p_i\|} \right\rangle$ as the cross angle of the point p_i

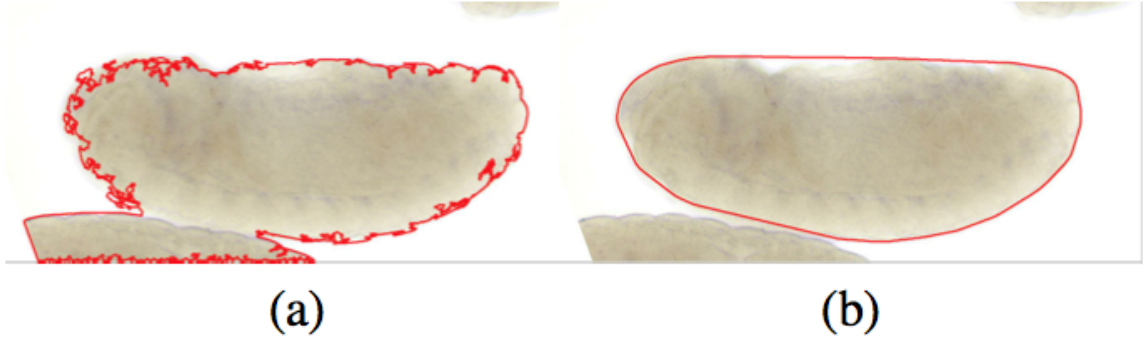


Figure 3.5: Effect of applying the angular checking constraint with $a_i \geq 135^\circ$

In Figure 3.5 both red boundaries indicate the largest connected component in each image before and after the angular constraint is applied. Figure 3.5(a) is both over-segmented and inaccurate relative to the proper location of the contour. The contour in 3.5(a) gets rejected because of the sharp corners caused by the touching embryo in the lower left corner. Figure 3.5(b) passes the angular check and we can see that the localization is much more precise than that found in 3.5(a).

One effect of introducing the angular checking constraint is that it ensures that the image has been smoothed as much as possible. It also has an effect such that when the merging occurs, it ensures that the smaller connected components have been merged and smoothed enough that they are relative to the true embryonic contour. In some way, the angular check itself essentially creates an upper bound simply from its implementation. If we observe the largest connected components of Figure 3.3 with $\sigma = [75, 100, 150, 200]$, the angular checking criterion would reject these from further analysis. Upon rejection, it

implies we will not need to spend the extra computation time required for comparison. The angular check guarantees a performance boost in computation time.

Border Checking

Recall that a challenge in embryonic contour extraction is distinguishing the validity of images that included single-touching, double-touching, or partial embryos near the image border. The angular checking constraint eliminated some of the failures caused from this phenomenon, but had difficulty dealing with most of them. There were some cases particularly near the image border where the framework mistakenly assumed that the partial embryo near the border was a part of the target embryo. Therefore, to aid in relieving this issue, a border checking constraint was implemented in an attempt to ensure that the selected scale image did not contain embryos from any angle the non-targeted embryo might be located.

The proposed border checking constraint involves examining the selected localized contour and if a given (x, y) pixel in the convex hull has either its x or y -value within this constraint from each side, then the entire boundary will be removed as a candidate. The border check is an adjustable parameter, but we found that leaving it static didn't cause the results to worsen. By using a border checking range that encompassed 3% of the original image size, this reduced many failed localizations caused by non-targeted embryos that were mistakenly targeted by the algorithm.

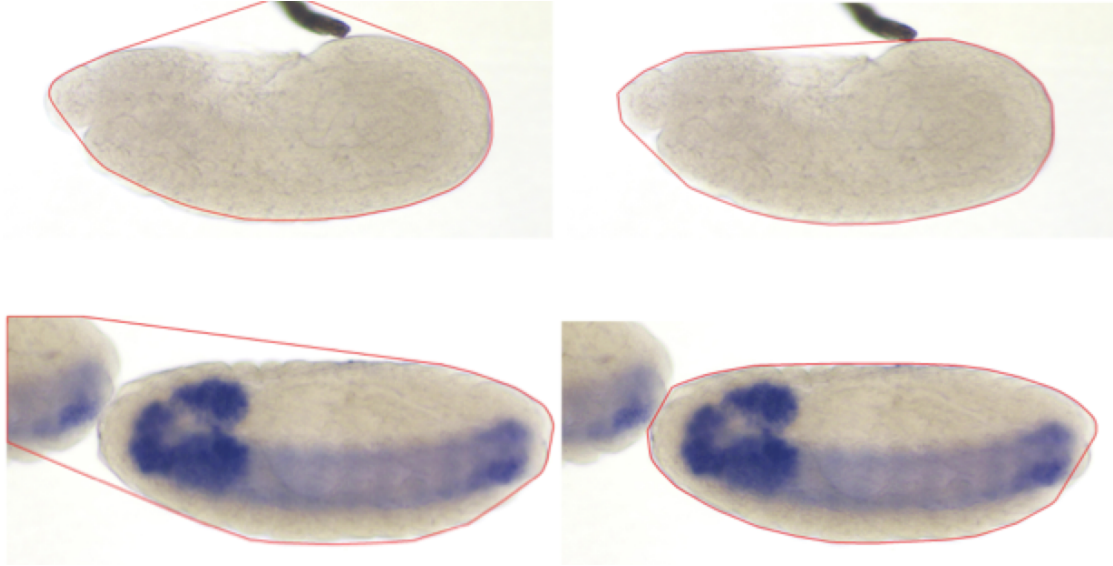
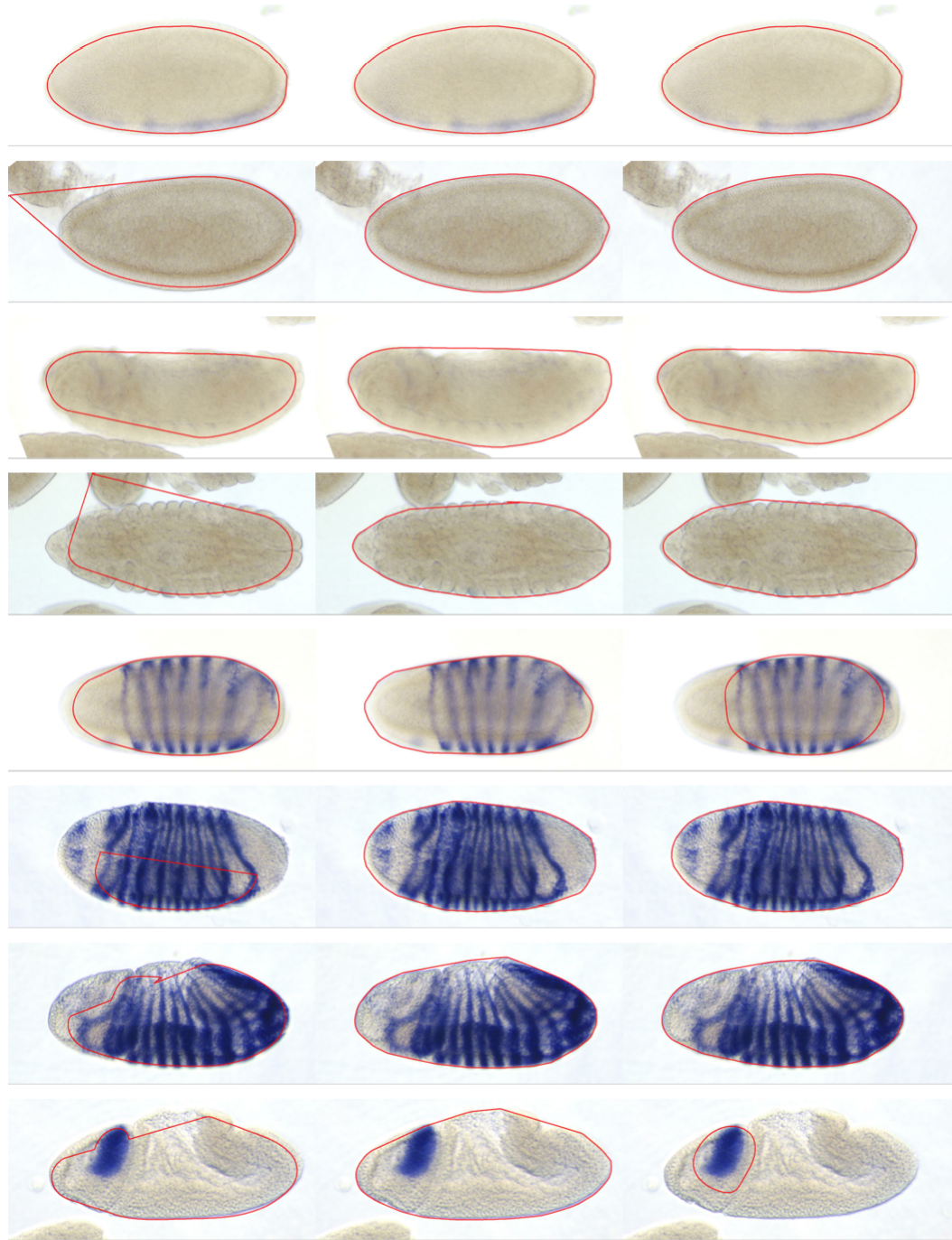


Figure 3.6: Sample images selected without border checking constraint (on left) with corrected image using the border checking constraint (on right).

Figure 3.6 demonstrates the effectiveness of the border checking constraint in addressing partial objects touching the border are rejected, resulting in a significantly more accurate localization. The removal of touching partial embryos played a large role in reducing inaccurate contours caused by touching embryos, partial embryos, or other features irrelevant to the targeted embryo.



Criterion 1

Criterion 2

Criterion 3

Figure 3.7: Effects of applying each criterion on embryonic images with varying conditions.

3.3 Results

In Figure 3.7, we demonstrate the effects of the framework after applying each of the criteria. The first four images are images that do not contain any expressive regions, however they do contain other potential environmental features, such as touching and partial embryos, that might have an effect on the localization. The lack of expressive regions makes the second and third criterion comparable to each other due to lack of high gradient within the targeted embryo. The last four images are embryos that contain expressive regions, some having much higher gradients than others. When we have to process a target embryo that has high inner gradient, the elliptical method fails as demonstrated in images 5 and 8. Though the shape modeling has been used successfully in previous research, we believe that it is unsuitable method based on the results from maximization of the largest connected component.

It is sensible that the largest connected component shows the most success due to the merging/diffusing effect of the Gaussian filter. There is also some convergence effect of using the Gaussian filter along with the non-maximum suppression of the Canny detector. In turn, it makes the largest connected component appear to be the most accurate localization criterion of the three.

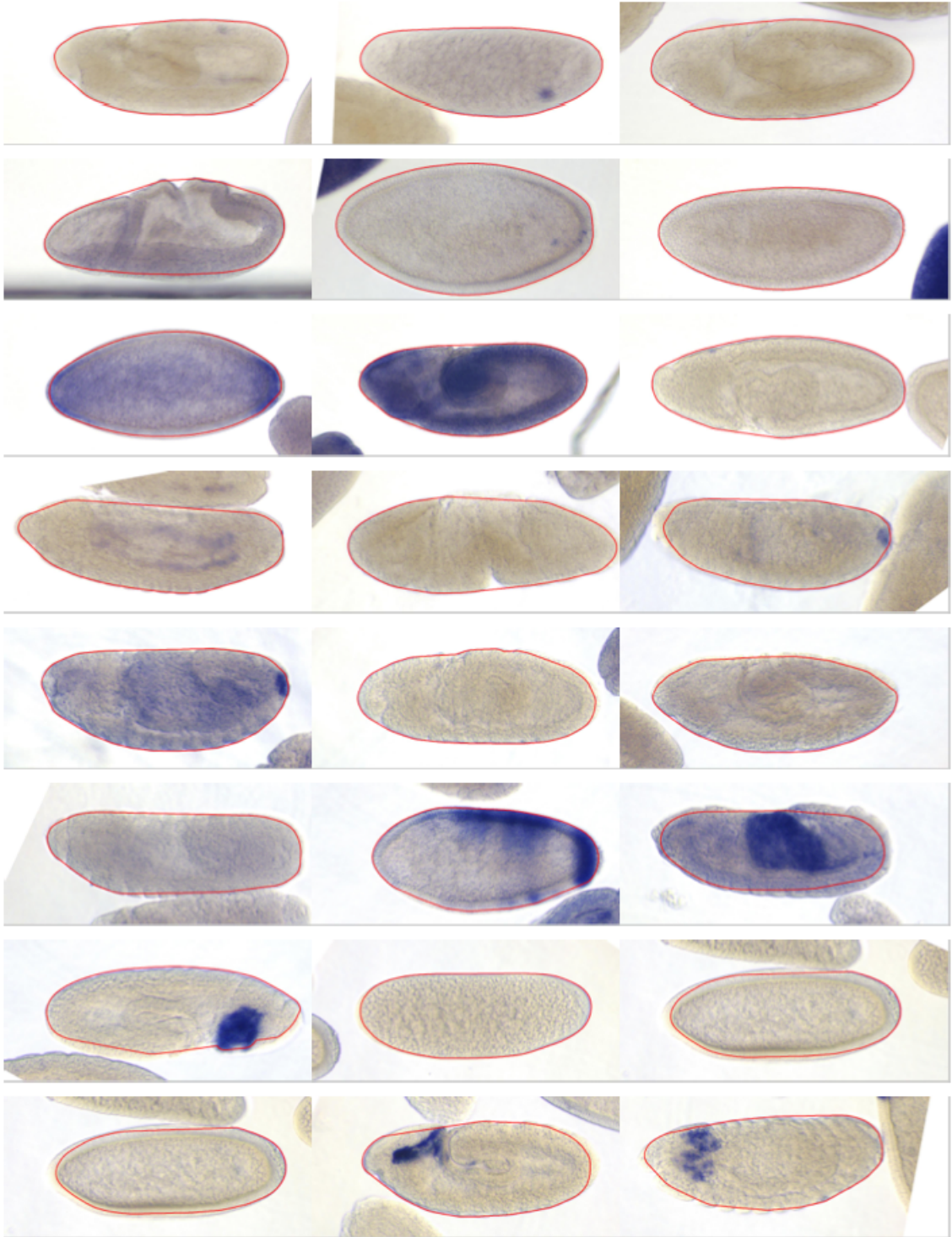


Figure 3.8: Successful localizations output from the framework.

The results proved overwhelmingly in favor in localizing the largest-of-largest component without the ellipse, achieving 91% accuracy when validating against ground truth data of manually clicked points where the proper boundaries are. Figure 3.8 demonstrates the effectiveness of our proposed framework. We also compared the results of the framework to the results from other research. If we do not use the scale space and simply apply the framework maximizing the largest connected component, we are only able to achieve a localization accuracy of 65%. Since active contour is widely used we also sought to compare our framework to the research that utilizes the active contour techniques. In comparison to the work done in Li et al [14], using the same dataset it achieved 86% accuracy. The framework’s accuracy appears to be superior to most other existing methods, however it does come with its limitations.

3.4 Conclusions

The proposed framework as defined by this paper has some limitations. Figure 3.9 demonstrates some of the notable failure cases from the framework. There were many issues that were caused by expressive region gradients being so strong that the framework was unable to localize the contour accurately. One such case is demonstrated in the first image from the left. The most common issue however was caused from partial and/or touching embryos with respect to the targeted embryo. For instance, the second and third cases of Figure 3.9 sufficiently demonstrate this effect, however with addition of the border checking constraint we were able to eliminate this issue.

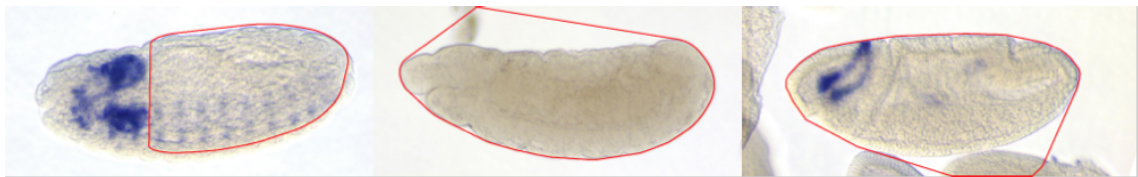


Figure 3.9: Failed localizations from the framework [4]

The framework also suffered in performance relative to the computational cost of the algorithm. We were unable to achieve fast computations due to the high-dimension images used, but changing the dimension could potentially affect the algorithmically determined localizations. We want the framework to achieve accurate results without requiring relatively the same computation time as some more advanced methods. This thesis aims to make significant improvements to the shortcomings of the proposed framework. One of the ways we seek to improve the framework is to reduce the scale parameter search space and take smaller steps, rather than simply subsampling to improve accuracy. We also seek to improve the efficiency of the framework.

IMPROVEMENT OF LOCALIZATION EFFICIENCY

In this chapter, we will propose a scheme to reduce the scale range in order to improve the localization efficiency.

4.1 Scale Range Reduction

4.1.1 Motivation

In the proposed framework, we subsampled scale space parameters on a non-continuous range from $[1, 200]$. By subsampling the scale range, it allowed us to see the general progression of the connected components across a large range of scales with relatively low computation time. We noticed a trend in the shapes and number of connected components in each value in the scale parameter range. Starting from $\sigma = 1$, the size of the contour has a tendency to increase until a certain peak. Prior to the peak, the numerous connected components begin to merge the several smaller components into a larger one. After reaching this peak, the largest component begins to distort. This effect is caused by the σ value increasing to such high values that it begins to suppress important contour information from the large component. The most accurate localizations tend to occur when the largest connected component reaches this peak value. Finding this peak value is non-trivial and will involve searching the scale space over a smaller interval.

From the results of the previous experiments we were able to conclude that:

- The optimal criterion of the three is the one that involves finding the largest-of-largest component over the scale space parameter range.
- The most valid range of scales is $[1, 40]$ Approximately 98% of the images respond best in this scale range.

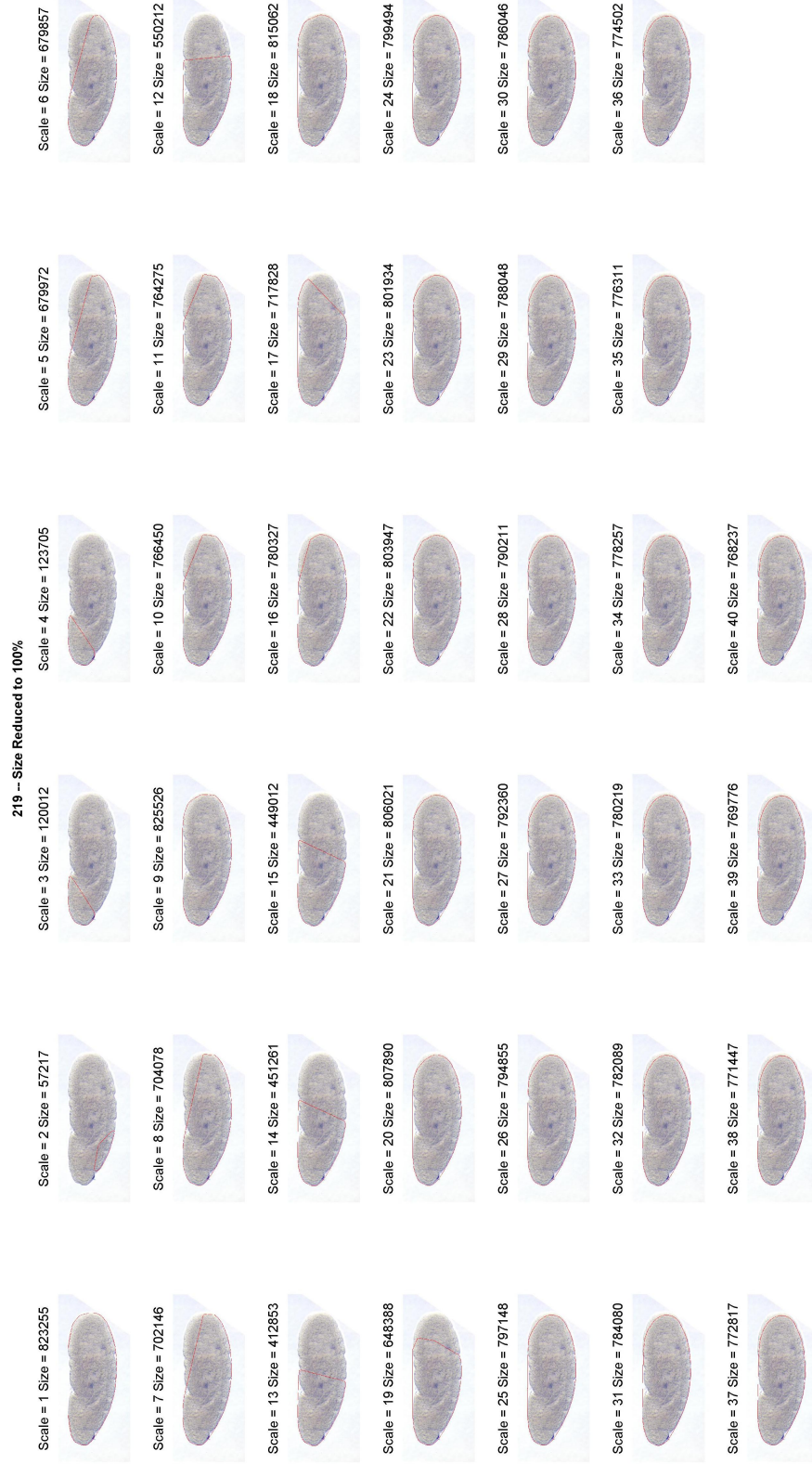


Figure 4.1: Framework results on a set of ISH images at original resolution with $\sigma \in [1, 40]$

In Figure 4.1, we show results of narrowing the scale range from $[1, 40]$. Since we've discovered a working scale parameter range for 98% of the image set, we should begin to narrow the range of values for extra precision. By modifying our scale space search range from $[1, 200]$ to $[1, 40]$, this gives us an 80% reduction from our original upper bound, which is a drastic improvement over our initial upper bound estimation. With such high accuracy within 20% of our original subsampled scale range from $[1, 200]$, it is safe to explore primarily on this range using smaller steps. However, since the lesser scale values were selected more often, it did not offer much of a decrease in the computation time we desired.

Limiting the step of our search will lead to a more appropriately localized contour, but there is a trade off in the computational cost of small steps. By using small values of σ , when the edge detection procedure of the framework performs its actions, it limits the blending of the gradient using the Gaussian blur. This in turn will leave many high-gradient discontinuous areas in the image that get translated into the edge image. Having too many components severely cripples the run time of the algorithm, so it is imperative to find the minimum components needed to perform an accurate localization.

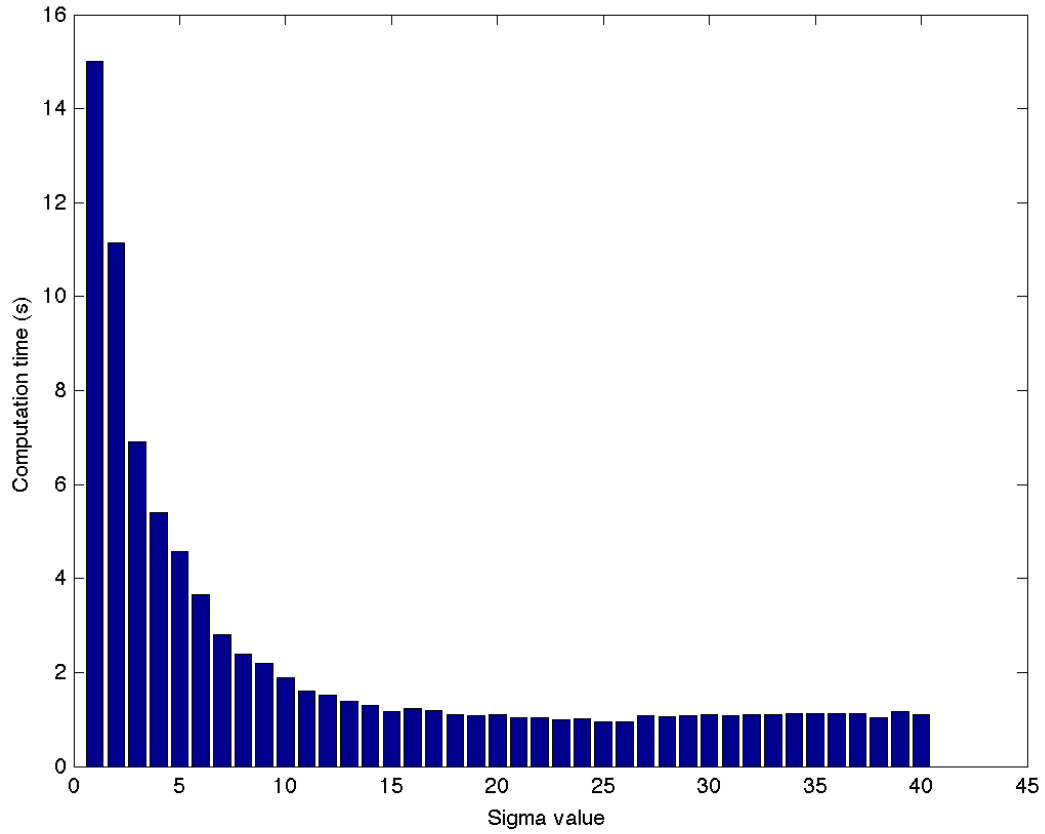


Figure 4.2: Computation time of a (raw) embryonic image exponentially decreasing with respect to scale values

Figure 4.2 shows the computation time for the framework on a given image using the proposed framework. For $\sigma > 15$, the computation time is relatively low due the number of connected components in the image being approximately five or less. As we increase the value of the scale parameter, we decrease the overall number of connected components because they will blur together. Using larger scale values results in many more small components occurring, thus leading to a dying exponential trend in computation time. The total computation time for this single image was 90 seconds. If we can select the optimal scale at $\sigma > 15$, then the computation time reduces in half.

Traditionally, the only way to find an optimal scale for a single image is to perform a linear search over the image convolved with each Gaussian filter over a

particular scale range. Since the experimental images in the BDGP dataset have a resolution of 2100×900 , it makes computing a large set of images very time consuming. A typical way to reduce the computation time of the images is to reduce their dimensionality. The BDGP ISH image set contains high-resolution images, so resizing to an easier computable stage would result in a drastic reduction in the computational cost of the algorithm. Figure 4.3 shows the result of applying the framework to the same image in different resolutions.

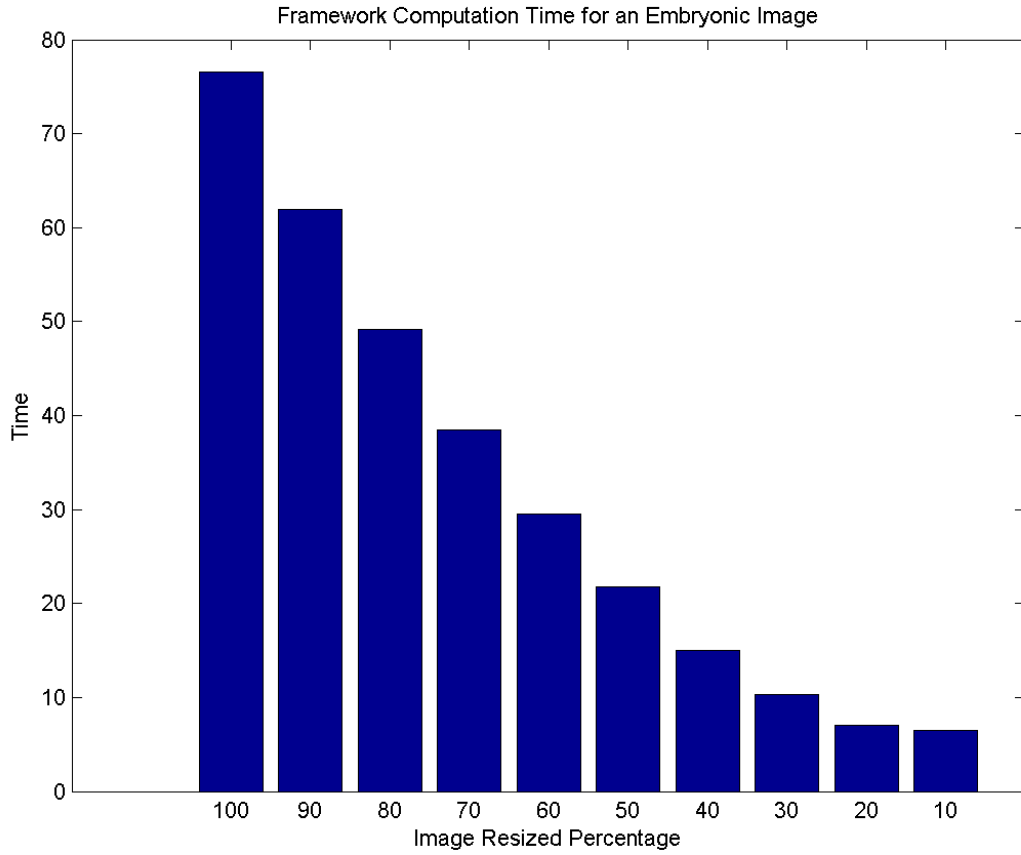


Figure 4.3: Computation time with respect to changes in resolution

We start from 100% of the original image size (2100×900) to 10% of the image size (210×90) with steps of 10%. The computation cost decreases exponentially with smaller and smaller resolution. It's important to consider the trade-off between

computational cost and localization accuracy between using low-resolution and high-resolution images. We want to be able to use the low-resolution images as a means to reduce the computation time, yet we want to use the high-resolution images to maintain the details that may have been lost from image size reduction.

In an effort to reduce the computation time and maintain localization accuracy, we propose using an inter-resolution scale mapping of low to high-resolution images. Given a high and low-resolution representation of an image (e.g., the raw image is high resolution and 10% resize of the image is low), we aim to find a correspondence between an optimal σ in the high-resolution image and an optimal σ in the low-resolution image. For the remainder of this work, we will define notation to represent the utilized parameters. Given an image, I , we denote $\sigma_L(I)$ to be an optimal scale with respect to the low-resolution representation of I . Similarly, we denote $\sigma_H(I)$ to be an optimal scale with respect to the high-resolution representation of I .

We denote σ_L to be the range of scales to explore within the low-resolution images. Similarly, we denote σ_H to be the range of scales to explore within the high-resolution (raw) images. The mapping scheme works by generating a $m \times n$ σ_L - σ_H table, where m is the naïve range of scales for low-resolution images and n is the naïve range of scales for high-resolution images. An element in the σ_L - σ_H table indicates the number of images such that $\sigma_L(I) = i$ and $\sigma_H(I) = j$.

Note that we are using a $m \times n$ table rather than a $n \times n$ dimension table. Since the scale parameter σ relates to the Gaussian filter window size, it's inappropriate to use the same range of values both on low and high-resolution images. In our experiments, we set a limit to the range, but try to make them relatively proportional with the resolution of the image. Had we not adjusted the low-resolution scale range, the contour would have appeared circular in shape and taken the form of the Gaussian filter rather than the target embryo shape.

4.2 Construction of the σ_L – σ_H Table

In our study, we set $m = 10$ and $n = 40$. To perform the resizing we used Bicubic interpolation. Bicubic interpolation computes the output pixel by averaging the nearest 4×4 neighboring pixels. We choose this type of interpolation for resizing because it leaves the image in a sharper, more natural state than to use other methods such as nearest-neighbor or bilinear interpolation. The convolution with the Gaussian filter will smooth out the image, and by adding extra smoothness *a priori* would have guaranteed us to lose more precision.

Algorithm 1 σ_L – σ_H table construction

Require: A set of images

Ensure: Table

```

for all Images in the set do
  Table[m][n]  $\leftarrow$  0
  for all  $\sigma_L$  do
     $\sigma_{L,optimal} \leftarrow framework(I, \sigma_L)$ 
  end for
  for all  $\sigma_H$  do
     $\sigma_{H,optimal} \leftarrow framework(I, \sigma_H)$ 
  end for
  Increment Table[ $\sigma_{L,optimal}$ ][ $\sigma_{H,optimal}$ ] by 1
end for

```

In Algorithm 1, we illustrate how we design the table. In the construction of the σ_L – σ_H table, we initialize a $m \times n$ table and its elements to zero. For every image in our image set, we apply the framework to generate the algorithmically chosen optimal σ using the largest-of-largest embryonic contour that passes all of the constraints. We use these optimal σ values as indices for the table and increment the location by one.

4.3 σ_L to $[\sigma_{H,min}, \sigma_{H,max}]$ Mapping

Given a σ_L , denote $\sigma_{H,min}$ the largest index such that $\forall j < \sigma_{H,min}, \text{Table}[\sigma_L][j] = 0$, and denote $\sigma_{H,max}$ the smallest index such that $\forall j > \sigma_{H,max}, \text{Table}[\sigma_L][j] = 0$. Once the σ_L - σ_H table has been completely built, we can examine the generated ranges to find a correspondence between the original, high-resolution contours versus the scaled, low-resolution contours. To find the range we examine each row, looking for the minimum and maximum columnar values greater than zero.

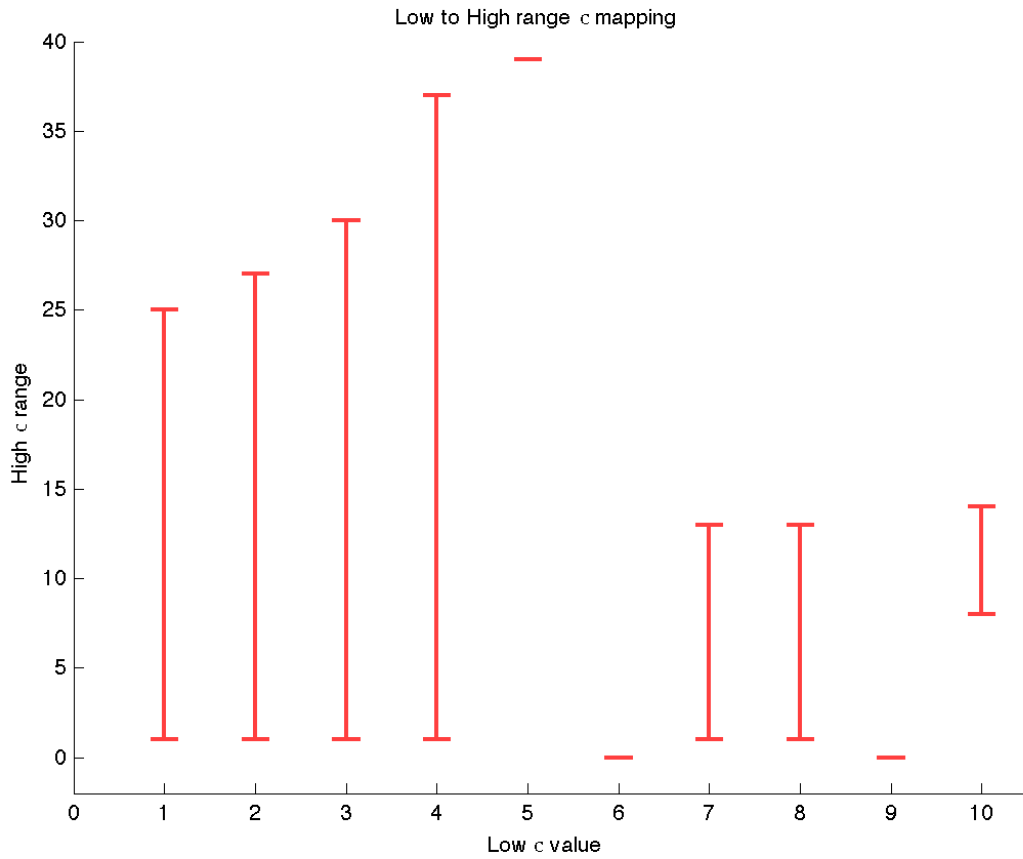


Figure 4.4: Table range of values for $\sigma_L \rightarrow [\sigma_{H,min}, \sigma_{H,max}]$

Figure 4.4 shows the range of $\sigma_L \rightarrow [\sigma_{H,min}, \sigma_{H,max}]$ extracted from the σ_L - σ_H

table. Quite possibly the first aspects noticed are that for $\sigma_L = 5, 6,$ and 9 either no range or a very limited range exists. For $\sigma_L = 6$ and 9 , there are no occurrences and $\sigma_L = 5$ has only one matching at $\sigma_H = 39$. When $\sigma_L > 4$, we see the range of optimal values decreasing when compared to $\sigma_L = [1, 4]$. On the lower range of σ_L , we see the upper bound increasing exponentially. We believe the exponential growth is attributed to the equation for the Gaussian filter itself, which uses σ^2 as the scale parameter. This finding leads us to believe that the best results must lie within the mapping where the range is relatively large. Note that Figure 4.4 does not reveal the actual number of occurrences and there is a possibility of a distribution lying within. Therefore, we should investigate the effect within this range of $\sigma_L = [1, 4]$.

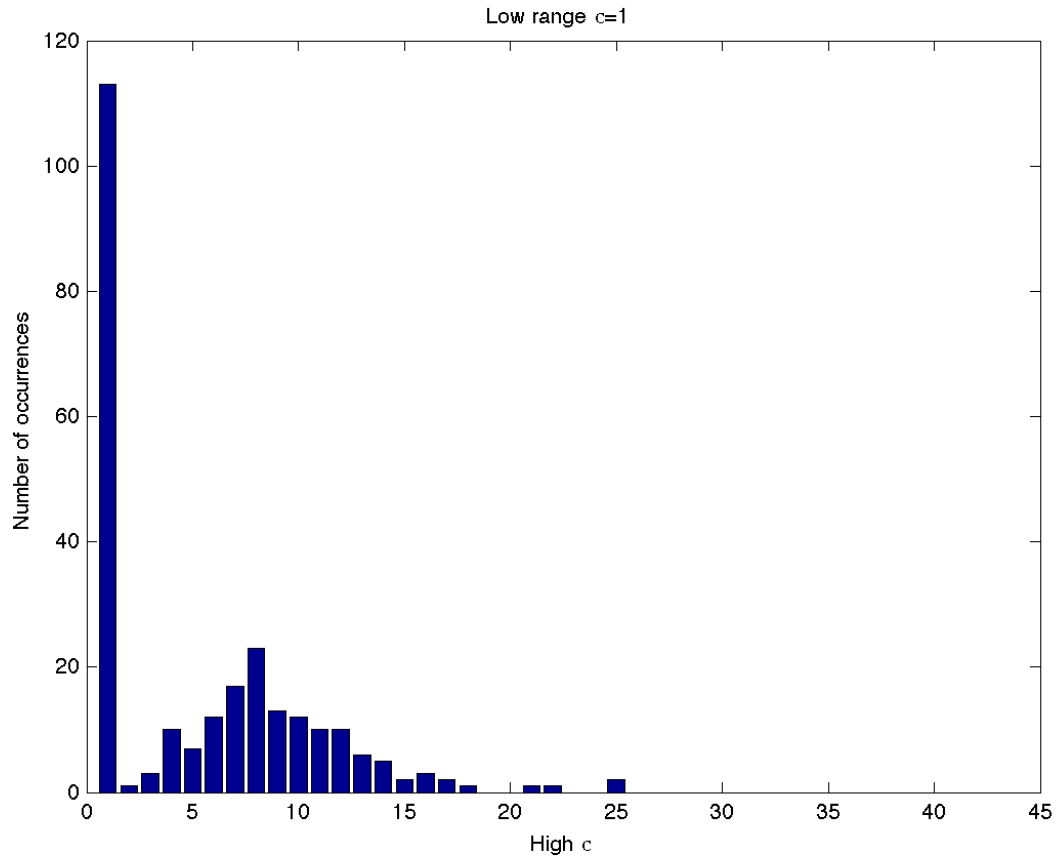


Figure 4.5: Table mapping distribution for $\sigma_L = 1 \rightarrow \sigma_H \in [1, 40]$

Upon inspecting the low-resolution scale range closer, we found that a large number of occurrences, about 86% of the image set, mapped from $\sigma_L = 1$. Figure 4.5 shows the mapping distribution particularly at this scale value since most of the optimal scales were found on that mapping.

Figure 4.5 shows an overwhelming amount of matches from $\sigma_L = 1 \rightarrow \sigma_H = 1$; approximately 38% of the total image set mapped in this manner. It doesn't appear to be very strange that the mapping occurred in this way. Due to the techniques that were involved, when we convolve an image and a Gaussian filter with $\sigma = 1$, the image's scale space representation is the nearly same as the image itself. The results then are the edges extracted from the detector processed through our algorithms and its optimality is determined under our constraints. Aside from $\sigma_L = 1 \rightarrow \sigma_H = 1$, the most interesting result of this is the distribution found from $\sigma_L = 1 \rightarrow \sigma_H \in [2, 40]$.

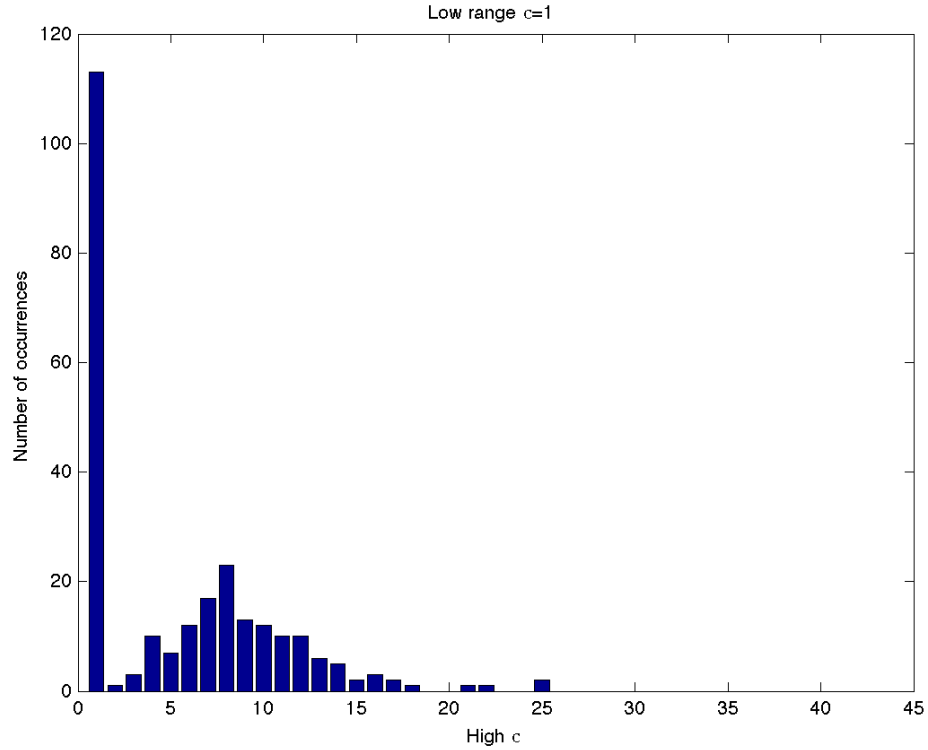


Figure 4.6: Table mapping distribution for $\sigma_L = 1 \rightarrow \sigma_H \in [2, 40]$

In Figure 4.7, we show the same occurrence mapping, excluding $\sigma_L = 1 \rightarrow \sigma_H = 1$. Approximately 48% of the image set mapped from $\sigma_L = 1 \rightarrow \sigma_H \in [1, 25]$. On first glance, we see that the data has a positive skew. Our initial ranges of high-resolution scales to explore were $[1, 40]$, yet the results from this mapping have no continuous occurrences after $\sigma_H = 18$. If we can isolate the scale range from $[1, 40]$ to $[2, 18]$, we can locate a Gaussian distribution within the mapping, leading us closer towards an optimal scale parameter. Since the basis of this experiment was to reduce the scale parameter search space, then by using this finding we can limit the mapping range appropriately to $\sigma_L = 1 \rightarrow [1, 18]$. This is a 55% reduction in the scale parameter search space, and based on the computation time extracted from Figure 4.2, equates to nearly a 30% reduction in computation time for each image to be processed.

4.4 Up-scaled Low-scale Contours Versus Original High-scale Contours

In Figure 4.2 we showed the computation time of the algorithm for localizing a contour in a high-resolution image. Since there is a significant difference in the computational cost for low-resolution images, we should try to utilize the results from the mapping and compare them to see which is a better fit. If the case where the up-scaled low-resolution contours fit better on the original sized image, then it would be appropriate to convert the image set to lower resolutions and extract from there.

We conducted an experiment to test the optimal boundary of the low-resolution image scales versus the high-resolution image scales. The validation involves minimizing the distance between the algorithmically selected contours at high and low scales versus the ground truth points to obtain minimal error. Given the low-scale algorithmically selected contour points, P_{L_i} , the high-scale algorithmically selected contour points P_{H_j} and the manually clicked ground truth points, Q_k , we calculate the point-to-point distance

as:

$$L_{i^*} = \operatorname{argmin}_k ||P_{L_i} - Q_k|| \quad (4.1)$$

$$H_{j^*} = \operatorname{argmin}_k ||P_{H_j} - Q_k|| \quad (4.2)$$

Where the optimal contour is selected by the minimal error function:

$$\min(\sum L_{i^*}, \sum H_{j^*}) \quad (4.3)$$

After performing the validation to the ground truth, we found that the algorithmically generated high-resolution points were preferred 58% of the time. Though it would have been more computationally convenient to use lower resolution images, the points were inaccurate with respect to the ground truth. The algorithm's preference to high resolution images comes from the loss of information when performing resolution reduction. Due to the interpolation method averaging the pixels and reducing to a smaller size, we lose much of the pixel locality that comes with analyzing high resolution images.

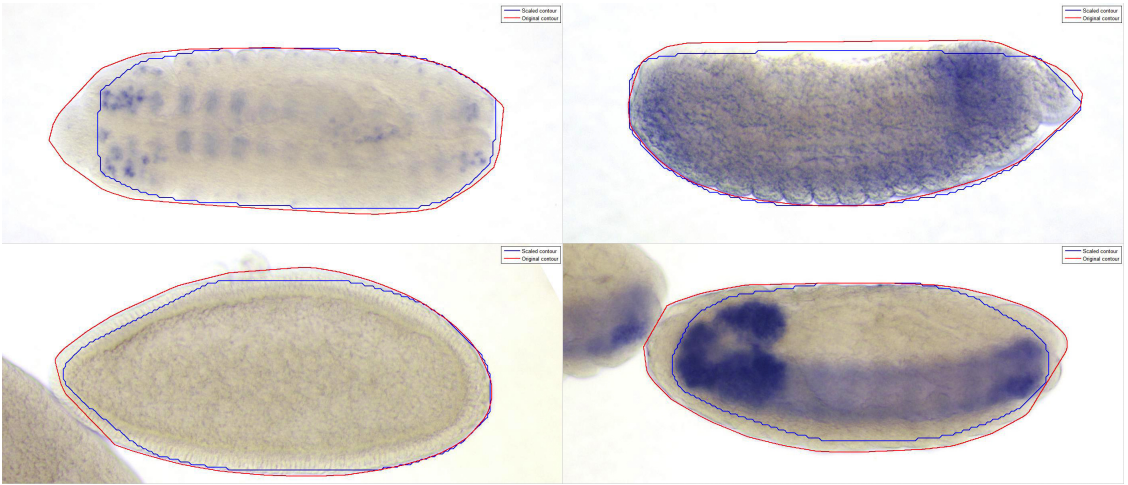


Figure 4.7: Embryonic image overlaid with low and high resolution contours.

Localizing contours by using low resolution images is significantly faster than on high resolution images. However, for work in the field of bioinformatics, precision is key. In the case of the *Drosophila* embryos, a localized contour that is off by a large factor could have devastating effects on the analysis of the embryo such as misjudging the stage or undetectable mutations that could have been found had the localization been more precise. The potential for selecting worse localizations cannot be allowed to propagate through the development of the framework, so it's imperative to always select the most reliable localization. Accurate localizations can later help determine the stages, mutations, and other gene-expressions in more advanced extensions of this framework such as implementing embryonic recognition or other high-level techniques.

CONCLUSION

The framework and the improvements that have been subsequently added to it since its inception have made it a very viable method to localize embryos in embryonic images. A variant of the framework has also been successfully applied to facial image localization and has performed substantially well when verified against the popular Viola-Jones method [13].

One of the issues currently surrounding the framework is that the algorithm can only account for convex polygonal shapes. Some embryos can have convex areas where the thorax of the embryo is beginning to develop in more advanced stages. In this case, the localization cannot account for concave regions. Since these potentially concave regions are not very deep, it may be possible to enhance the localization precision by using an active contour to more precisely represent the embryonic shape. Other techniques such as gradient vector flow (GVF) [12] may be applied in the future to further improve the localization precision. For embryos that are in a low level stage of development, the concavity is not an issue because they haven't been able to develop any features yet. Only once the embryo is in a later developmental stage does this ever present itself as an issue.

One lingering difficulty still lies in reduction of the computational cost. We were able to reduce the scale parameter search space, but this does not reduce the computational cost of the framework itself. For performance issues, we can balance the accuracy versus efficiency tradeoff by adjusting the upper and lower bounds in the mapping scheme. Later experiments proved that keeping the upper bound of the image size greater than 75% proved to be able to maintain accuracy and reduce the computation time. It also may be possible to increase the lower bound due to the little difference between using a low-resolution image at 10% versus 20% of the original image size.

Overall, we have introduced a framework that offers substantial advantages because it is completely automated and highly accurate. The fact that human interaction is not necessary to achieve these results is an attractive feature that makes this framework

both academically and industrially practical. The framework achieves high-level accuracy by using primarily low-level techniques such as Gaussian smoothing and edge detection over various embryonic conditions. This framework has great promise to be extended and applied to other variations of images to automatically localize a region of interest.

BIBLIOGRAPHY

- [1] Berkeley *Drosophila* Genome Project. <http://www.fruitfly.org/>.
- [2] Yusuf Sinan Akgul and Chandra Kambhamettu. A scale-space based approach for deformable contour optimization. In *Proceedings of the Second International Conference on Scale-Space Theories in Computer Vision, SCALE-SPACE '99*, pages 410–422, London, UK, UK, 1999. Springer-Verlag.
- [3] Soujanya Siddavaram Ananta. Contour Extraction of *Drosophila* Embryos Using Active Contours in Scale Space, 2012.
- [4] Z. Bessinger, Guangming Xing, and Qi Li. Localization of *drosophila* embryos using connected components in scale space. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 497–500, 2012.
- [5] S. Beucher and F. Meyer. The morphological approach to segmentation: the watershed transformation. Mathematical morphology in image processing. *Optical Engineering*, 34:433–481, 1993.
- [6] John Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.
- [7] Charles DeLisi. The Human Genome Project. *American Scientist*, 76(488), 1988.
- [8] A.W. Fitzgibbon, M. Pilu, and R.B. Fisher. Direct least squares fitting of ellipses. In *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, volume 1, pages 253–257 vol.1, 1996.
- [9] M. Gargesha, J. Yang, B. Van Emden, S. Panchanathan, and S. Kumar. Automatic annotation techniques for gene expression images of the fruit fly embryo. In S. Li, F. Pereira, H. Y. Shum, and A. G. Tescher, editors, *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 5960 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 576–583, July 2005.
- [10] Janssens H, Kosman D, Vanario-Alonso CE, Jaeger J, Samsonova M, et al. A highthroughput method for quantifying gene expression data from early *drosophila* embryos. *Development, Genes and Evolution*, 215:374–381.
- [11] Michael Kass, Andrew P. Witkin, and Demetri Terzopoulos. Snakes: Active Contour Models. *International Journal of Computer Vision*, 1(4):321–331, 1988.
- [12] Satyanad Kichenassamy, Arun Kumar, Peter Olver, Allen Tannenbaum, and Anthony Yezzi Jr. Gradient flows and geometric active contour models, 1994.
- [13] Qi Li and Zachary Bessinger. Learning scale ranges for the extraction of regions of interest. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 2581–2584, 2012.

- [14] Qi Li and Chandra Kambhamettu. Contour extraction of drosophila embryos. *IEEE/ACM Trans. Comput. Biology Bioinform.*, pages 1509–1521, 2011.
- [15] Tony Lindeberg. Edge detection and ridge detection with automatic scale selection. In *CVPR*, pages 465–470, 1996.
- [16] Tony Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):79–116, 1998.
- [17] Daniel L. Mace, Nicole Varnado, Weiping Zhang, Erwin Frise, and Uwe Ohler. Extraction and comparison of gene expression patterns from 2d rna *in situ* hybridization images. *Bioinformatics*, 26(6):761–769, 2010.
- [18] Morales DA, Bengoetxea E, and Larranaga P . Automatic segmentation of zona pellucida in human embryo images applying an active contour model. In *Proceedings of the 12th Annual Conference on Medical Image Understanding and Analysis*, pages 209–213.
- [19] C Niehrs N Pollet. Expression profiling by systematic high-throughput *in situ* hybridization to whole-mount embryos., 2001.
- [20] R. Rajesh N. Senthilkumaran. Edge detection techniques for image segmentation and a survey of soft computing approaches, 2009.
- [21] Jia-Yu Pan, André G. R. Balan, Eric P. Xing, Agma J. M. Traina, and Christos Faloutsos. Automatic mining of fruit fly embryo images. In *KDD*, pages 693–698, 2006.
- [22] Hanchuan Peng and Eugene W. Myers. Comparing *in situ* mrna expression patterns of drosophila embryos. In *Proceedings of the Eighth Annual International Conference on Research in Computational Molecular Biology, RECOMB '04*, pages 157–166, New York, NY, USA, 2004. ACM.
- [23] Chenyang; Prince Jerry L Pham, Dzung L.; Xu. Current Methods in Medical Image Segmentation. pages 315–337.
- [24] Weizmann R Kwan E Shu S Lewis S Richards S Ashburner M Hartenstein V Celniker S. et al. Tomancak P, Beaton A. Systematic determination of patterns of gene expression during drosophila embryogenesis. *Genome Biol.*, 3(12).
- [25] Guanglei Xiong, Xiaobo Zhou, and Liang Ji. Automated segmentation of drosophila rna*i* fluorescence cellular images using deformable models. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 53(11):2415–2424, 2006.
- [26] Chenyang Xu and Jerry L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing*, 7(3):359–369, 1998.

